Docket No.: 50277-1747 PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of	) Confirmation No. 1591
Steven Viavant, et al.	) Examiner: Kamal B. Divecha
Application No. 09/945,160	) Group Art Unit: 2151
Filed: August 31, 2001	)
For: TECHNIQUES FOR SERVER-CONTROLLED MEASUREMENT OF CLIENT-SIDE PERFORMANCE	)

DECLARATION UNDER 37 CFR 1.131 OF STEVEN VIAVANT

I, Steven Viavant, declare that:

1. I am the first named inventor of the present application.
2. The invention recited in the claims of the subject patent application was actually reduced to practice in various versions of prototype software for Oracle Corporation's IAS software between December 15, 2000 and January 9, 2001, as indicated in the attached invention disclosure form that describes the invention claimed in the present patent application.
3. The attached invention disclosure form does not include information about dates of conception. The invention recited in the claims of the subject patent application was, however, conceived and discussed among the co-inventors of the present patent application at least two days prior to the earliest actual reduction to practice of December 15, 2000, or at least as early as December 13, 2000.
4. From a time just prior to December 14, 2000 to the actual implementation of the invention in prototype software between December 15, 2000 and January 9, 2001, the inventors worked diligently to implement the invention recited in the claims of the present application in the prototype software.
6. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that

these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Date: 1/3/08

Steven Viavant

Wrt to file
by 1/24/01

[redacted] Patent Operations

Oracle Corporation
500 Oracle Parkway
M/S 5op7
Redwood Shores
California 94065

Phone 650.506.5600
Fax 650.506.7114
patents@us.oracle.com

FACSIMILE COVER SHEET

Date: **Wednesday, January 17, 2001**

To: **Brian D. Hickman**

Fax: **(408) 414-1076**

Hickman Palermo Truong & Becker
1600 Willow Street
San Jose, CA 95125
Phone: (408) 414-1080

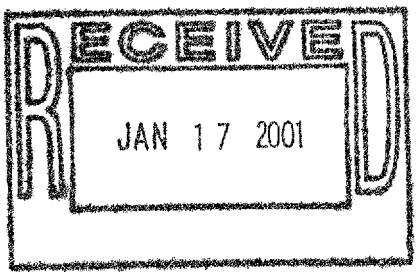
From: **Kathy Farrell (650) 506-8441**

Pages: **5 (Including cover)**

Re: Patent Application
Title: METHOD FOR WEB CLIENT RESPONSE-TIME MEASUREMENT
Inventors: Steven Viavant; Arsalan Farooq; Jaydeep Marfatia; Manu Shukla
Oracle Matter: OID-2001-001-01
Firm File #: unknown

(650) 506 2913

Message: Hi Brian, I understand from Michael that a provisional will be filed on the enclosed invention disclosure.



[Fwd: OID-2001-001-01

Return-Path: <kathleen.a.farrell@oracle.com>
Received: from oracle.com (dhcp-5op6-5op6-west-144-25-178-224.us.oracle.com [144.25.178.224])by gmgw01.oraclecorp.com (8.8.8 .8.8) with ESMTP id LAA15616for <kathleen.a.farrell@oracle.com>; Wed, 10 Jan 2001 11:24:51 -0800 (PST)
Message-ID: <3A5CB7D7.E3DA4A65@oracle.com>
Date: Wed, 10 Jan 2001 11:28:23 -0800
From: Kathleen Farrell <kathleen.a.farrell@oracle.com>
Organization: Oracle Corporation
X-Mailer: Mozilla 4.61 [en] (Win95; U)
X-Accept-Language: en
MIME-Version: 1.0
To: Kathleen Farrell <kathleen.a.farrell@oracle.com>
Subject: **[Fwd: OID-2001-001-01**
Content-Type: multipart/mixed;boundary="----------11845F4556F013509C9D1B7A"

OID-2001-001-01
* *

Regards,

Kathy Farrell
Senior Patent Administrative Asst.
Kathleen.A.Farrell@oracle.com
Oracle Corp. 5op717a
phone: (650) 506-8441
fax: (650) 506-7114

Return-Path: <legal@us.oracle.com>
Received: from web18.us.oracle.com (web18.us.oracle.com [130.35.133.100])by gmgw03.oraclecorp.com (8.8.8 .8.8) with ESMTP id SAA19518for <kathleen.a.farrell@oracle.com>; Tue, 9 Jan 2001 18:49:57 -0800 (PST)
Received: (from legal@localhost)by web18.us.oracle.com (8.8.8 .8.5) id SAA16474;Tue, 9 Jan 2001 18:49:56 -0800 (PST)
Date: Tue, 9 Jan 2001 18:49:56 -0800 (PST)
Message-ID: <200101100249.SAA16474@web18.us.oracle.com>
To: kathleen.a.farrell@oracle.com
From: New-inventor_us@oracle.com
Subject: **WEB INVENTION DISCLOSURE FORM**

Descriptive Title or Subject of the Invention:

Method for web client response-time measurement

Inventors:

Steven Vlavant, 90 Crocker, Piedmont, Ca 94611, (510) 420 8449,
manager is Geoff Wilson
Arsalan Farooq, 2618 Greenwich St., Apt #1 San Francisco, CA 9412 (415) 346 2926,
manager is Steven Vlavant
Jaydeep Marfatia, 3292 Colgate Ave, Santa Clara CA 95051, (408) 248 8963, manager is Steven Vlavant
Manu Shukla, 10 De Sabla Road Apt 304, San Mateo, Ca, 94402, (650) 342-6446, manager is Steven Vlavant

[Fwd: OID-2001-001-01

Primary Development Group: ST-HQ
Secondary Development Group: System Management Products

Managing VP: Jay Rossiter
Other VP:

Description of problem:

People use browsers on the net to effect all sorts of useful activities. Providers of web sites are interested to know how quickly their site is serving content to the people who come to it.

One way of doing this is to measure at the web site itself how quickly one is able to serve requests. However, this does not tell you what the 'customer' is really experiencing, since there are additional delays associated to network time, as well as for the client (e.g. PC) to actually render the content on its screen.

Some companies try to approximate this measurement by running a set of distributed agents at different points on the internet, and having them periodically simulate a customer by downloading one or several specific html pages. This has the drawback that it only measures a few of the different activities (pages) on the site, and that although it measures these times from a variety of geographic locations, it doesn't measure them from the actual locations the customers are coming to the site from. And, it imposes gratuitous load on the system in the process, plus requires a separate out-of-band step to ship the measurements to the site which actually is interested in them.

Another approach is to install some software on the client PC to measure these times. This approach has been taken variously by installing a specially instrumented replacement tcp stack, by installing a special device driver which maps in the memory of other device drivers and snoops on them, etc. This approach has the drawback that installing (and maintaining) software on the client is a very cumbersome thing, and has all sorts of management headaches associated with reporting the measurements obtained across the internet, etc..

Yet another approach is to instrument the application (content) itself to take the measurements and report the results. This has the drawback that potentially thousands of separate instrumentation points need to be inserted and maintained in the code/content that constitutes the application.

Description of the solution of the invention:

Our approach is to automatically instrument the content of the application using some generic javascript, which we append to the content as it is being served. This javascript is designed so that it may be appended to any HTML page, at the end of the page. It registers 'handlers' for certain key events, such as a page finishing rendering, or a user clicking on a submit button, etc. We use these handlers to capture client clock values, which we then automatically transmit back to the web site which served the original content, either using a cookie, by using the argument field of a fetch of a dummy (non-displayed) image, or using an html POST operation (each method has certain advantages over the others). This data is then mined from log files at the web server side (or in the case of using the POST operation, handled directly by a servelet).

The automatic instrumentation can be effected in various manners. The existing prototype invention uses a "vanilla" Apache "mod". Other approaches investigated (and in various

stages of implementation) include using custom code inside the Oracle webcached, or altering the base servelet of the servelet engine being used.

The pros and cons of the invention over status quo:

See parts A and B.

Significance of the new solution to Oracle:

Any web site which deploys the Oracle stack will be able to obtain accurate, "click to eyeball" response time values for all of the content it serves, for any client, anywhere on the internet, for any application, without installing anything on the clients, or manually instrumenting their content. To our knowledge, nobody else can make this claim currently.

This sort of info is valuable for providing Service Level Reports for hosting environments, which can serve as the basis for monitoring Service Level Agreements (especially in the case that there is a VPN or other network component involved, which the Application hosting entity takes responsibility for).

This sort of information is also valuable in such an environment, in that it allows a service provider to have a window into the 'shop' of its customers, and enables reports to the customer as to which users should have their PC or network connection investigated for performance problems (i.e. we can look at a set of data, and see that repeatedly, at time T joe had response time of 42 seconds for an activity that most users received at time T in 7 seconds; we don't know what joe's problem is, but we know that he has one. This allows joe's sys-admins to investigate and remedy the situation; also note that we can prioritize this list of "joes" by those whose productivity is most affected).

This is also a good mechanism for finding content with pathological rendering problems, or problems relating to certain browser type or browser version only (and to ignore such potential problems if your actual user base is not using the browser type or version which would trigger the problem).

This invention could also be used to find out which clients have problems with apps with large or complicated content, and could be used to drive an automated 'adaptive' choice of simple vs. complex versions of the same content to serve, depending on apparent (measured) client capacity.

This technique could also provide data to trending calculations, to obtain advance notice of gradual deterioration of response times, as well as to drive 'events' (e.g. email or pagers) in the event of sudden, steep degradation in response time, to alert a sys-admin to respond (or to automatically begin shedding or redirecting load, or serve the 'lightweight' version of content, etc).

This sort of data could also be used to debug complex site administration problems, by narrowing the time frame to those slots which actually had a negative affect on customer performance. For example, a site typically gathers a large number of metrics (cpu, memory, disk, database-buffer-cache, etc). There may easily be upwards of hundreds of such metrics monitored on the different tiers and stacks deployed at a site. Each of these metrics may at various times spike or exceed 'comfortable' thresholds. However, some of the time this may have no affect on the end-users of the site, other times it may have grave ill effects on them. By being in possession of the data telling us specifically when customers had problems, we can mine our other system management data in an intelligent way, and focus our attention quickly on the particular metric-threshold-violations which were likley to have contributed to real problems, as opposed to those which were benign. [which if we automated m!

ight in fact be the basis for a separate disclosure...].

And in general, it is just nice to get a handle on the response time that your real customers are actually experiencing.

Does it add or enhance functions or features? Yes
Does it increase performance for existing functionality? No
Does it pertain to an interface? No
Is it detectable by an end user, DBA, or SysAdmin? Yes

Is it applicable to products or environments outside of Oracle?

The same technique could be used in conjunction with any Web server, not only with Oracle IAS.

Has it been implemented? Yes
If so, when? prototypes with various degrees of completeness over the period 12/15/00 to present.
Has it been disclosed outside of Oracle? No
If so, was it disclosed under a Non-Disclosure Agreement?

Description of any disclosure outside of Oracle:

Description of any products that use the invention, currently or planned:

Anticipated with 9i IAS release 2, approximate dates of Beta 5/01, production 9/01. Possible candidate for release in patch version of 9i IAS release 1, before then.

Kathleen Farrell <kathleen.a.farrell@oracle.com>